8. National Conference on Computational Mechanics
MekIT'15
B. Skallerud and H I Andersson (Eds)

TOWARDS A CONTINUOUS AND DISCONTINUOUS GALERKIN METHOD FOR MULTI-PHASE FLOWS

TORMOD LANDET, MIKAEL MORTENSEN, JOHN GRUE

Department of Mathematics
Faculty of Mathematics and Natural Sciences, Universtity of Oslo
Niels Henrik Abels hus, Moltke Moes vei 35, Oslo, Norway
Corresponding author: Tormod Landet, tormodla@math.uio.no

Key words: finite elements, discontinuous Galerkin, two-phase flow, free surface

Abstract. We present an overview of the continuous and discontinuous Galerkin finite element methods (CG-FEM, DG-FEM) for numerical solution of the transport equation and show how the Navier-Stokes equations can be solved with CG-FEM in both a coupled and segregated manner. We show convergence of the implemented Navier-Stokes solvers for single-phase flows on the well known Taylor-Green vortex and also show how the volume of fluid (VOF) method for tracking the interface between two fluids can be implemented in DG-FEM. Our work is aimed towards creating a method for looking at the impact forces from water waves on structures and comments are made on how to evolve the presented solution methods into a two-phase solver that can handle the physics involved in highly non-linear free surface simulations.

1 INTRODUCTION

Numerical simulation of two-phase flows with the Navier-Stokes equations and surface capturing techniques is an active field of research. In the industry computational fluid dynamics, CFD, applications are increasingly being used to study large two-phase flow problems, such as ocean waves around fixed and floating structures. The industry standard CFD tools for free surface physics are to a very large extent dominated by finite volume methods for solution of the Navier-Stokes equations and two phase flow models based on the volume of fluid, VOF, method for capturing the free surface as an immersed interface which is transported with the global flow field. The main reasons for the popularity of this combination is the ease of handling complicated geometries with unstructured meshes, the robustness and generality of the free surface capturing technique and the good mass conservation properties.

The main alternatives to the finite volume method for solving the Navier-Stokes equations on unstructured meshes are the continuous and discontinuous Galerkin finite element

methods. Both CG-FEM and the finite volume method can be seen as special cases of the more general discontinuous Galerkin method. The main advantage of the DG-FEM method is the possibility to have local high order approximating polynomials while maintaining local conservation. The main disadvantage is the increased number of unknowns. [10]

This work presents a step on the way to create a two phase Navier-Stokes solver utilising the advances in the finite element family of methods over the last decades. The work consists of a presentation of the building blocks of continuous and discontinuous Galerkin formulations of the advection equation and a continuous Galerkin formulation of a Navier-Stokes equations. Two Navier-Stokes solution methods using the continuous formulation are presented and results in form of convergence rates are shown. Furthermore a volume of fluid method using the discontinuous Galerkin method to discretise the advection equation for the colour function is presented. The results are compared for one naive and one more advanced convection scheme for the VOF colour function.

2 THE GOVERNING EQUATIONS

The Navier-Stokes equations on conservative form are given as follows for a velocity field $\mathbf{u}$ and pressure field p with density ρ and dynamic viscosity μ:

$$\frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = \nabla \cdot \mu \left[(\nabla \mathbf{u}) + (\nabla \mathbf{u})^T \right] - \nabla p + \rho \mathbf{g}, \tag{1}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0. \tag{2}$$

We will use a decoupled volume of fluid formulation with two incompressible phases where volume conservation implies mass conservation so equation (2) will turn into the divergence free criterion, $\nabla \cdot \mathbf{u} = 0$, when solving for the unknown velocity and pressure. The VOF formulation is described in section 4.

The Navier-Stokes equations can either be solved directly as a coupled system of equations, or as a decoupled system of equations that approximates the coupled system. The advantage of the decoupling is that the resulting linear equation systems resulting from the discretisation of the equations on a computational mesh are smaller, and fast iterative Krylov subspace methods with appropriate preconditioners are known for these decoupled equations. For the coupled system it is more common to use direct solvers which alleviates the need to find a good preconditioner, but has the disadvantage of taking more time for large distributed calculations.

A very common method to decouple the governing equations is the pressure correction/projection method pioneered by Chorin [4] and Temam [21]. The decoupling can be performed such that the correction procedure is performed iteratively for each time step [8]. This leads to better agreement with the coupled solution and higher order convergence in space and time can be achieved [23] with this type of incremental pressure correction scheme.

The implemented Navier-Stokes solver, named Ocellaris [15], developed for the work described in this article contains both a coupled solver and a standard incremental pressure correction scheme decoupled solver. Both solvers exhibit the same convergence rates in the L_2 and H^1 norms on constant density flow benchmarks like the Taylor-Green vortex as will be shown later.

A second order IPCS scheme written using a second order backwards difference formulation, BDF2, decouples the equations as follows:

$$\frac{1}{2\Delta t}\left[3(\rho\mathbf{u})^* - 4(\rho\mathbf{u})^k + (\rho\mathbf{u})^{k-1}\right] + \nabla\cdot\left[(\rho\mathbf{u})^*\otimes\mathbf{u}^C\right] = \nabla\cdot\mu\left[(\nabla\mathbf{u}^*) + (\nabla\mathbf{u}^E)^T\right] - \nabla p^* + \mathbf{g} \tag{3}$$

$$\nabla\cdot(\text{eq. 5});\ \nabla\cdot\mathbf{u}^{k+1} = 0 \qquad \Rightarrow \quad \nabla\cdot\frac{1}{\rho}\nabla\hat{p} = \frac{3}{2\Delta t}\nabla\cdot\mathbf{u}^* \tag{4}$$

$$\frac{1}{2\Delta t}\left[3\mathbf{u}^{k+1} - 3\mathbf{u}^*\right] = -\frac{1}{\rho}\nabla\hat{p} \tag{5}$$

The momentum equation (3) is solved for a tentative velocity $\mathbf{u}^*$. A pressure correction is calculated from the pressure Poisson equation (4) which is formulated by taking the divergence of the Helmholtz-Hodge decomposition [14] of the velocity into a solenoidal (divergence free) velocity field and the gradient of a scalar field. The pressure is then updated $p^{k+1} = p^* + \hat{p}$ and this new pressure can either be used in a new evaluation of the momentum equation, or used to update the velocity at the end of the time step by equation (5).

To linearise the equations we will use a second order backwards difference extrapolation of one of the velocity terms in the convection term in the momentum equation to the $k+1$ time step. This will be used both in the coupled and decoupled equation systems.

$$\mathbf{u}^C = 2\mathbf{u}^k - \mathbf{u}^{k-1} \tag{6}$$

To fully decouple the velocity components in the momentum equations we need to make the second part of the diffusion term explicit. This term is small, since the velocity field is almost solenoidal/divergence free in our iterations. We find that using $\mathbf{u}^E = \mathbf{u}^k$ gives second order convergence in time on the Taylor-Green vortex test case. Taking $\mathbf{u}^E = \mathbf{u}^C$ leads to oscillatory results and does not converge.

3 VARIATIONAL FORMULATION

3.1 Preliminaries

The process of formulating the variational, or weak, form of the governing equations (1) and (2) consists of treating each term in the equations by itself. Below we treat term by term in the equation and explain how the variational formulation is derived for the continuous Galerkin method. For the discontinuous Galerkin method the derivation of

some terms is a bit more involved, and only terms that are necessary formulate an unsteady advection equation will be discussed.

We start with a computational domain $\Omega \in \mathbb{R}^D$ with boundary $\partial\Omega$. This domain is divided into a tessellation $\mathcal{T}$ of non-overlapping elements $\mathcal{K}$ such that $\mathcal{T} = \bigcup_i \mathcal{K}_i$. Each element $\mathcal{K}$ has a boundary $\partial\mathcal{K}$.

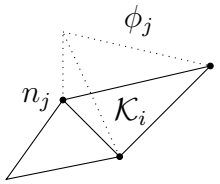

Figure 1: A linear discontinuous nodal basis function ϕ_j at node n_j

The size of an element is characterised by its characteristic length h and the element has associated polynomial basis functions of order P. These basis functions are nodal which means that the value of the function is unity at the associated node and zero at all other nodes. The basis functions are continuous across element boundaries in the continuous Galerkin method, CG-FEM, and discontinuous across element boundaries in the discontinuous Galerkin method, DG-FEM. This means that for each node in the mesh we have one associated degree of freedom in CG-FEM while in DG-FEM we get one degree of freedom for each element that shares the node.

In the discontinuous Galerkin setting we will need to address the discontinuous solution at a shared face $\mathcal{F}$. We will look at the two elements that share the face $\mathcal{F}$ and following the normal discontinuous Galerkin nomenclature we denote one element $\mathcal{K}^+$ and the neighbour $\mathcal{K}^-$. We are working on unstructured meshes so there is no inherent geometrical ordering that causes one element to carry the $+$ and the other the $-$; the naming is arbitrary. Both elements have associated outwards pointing normal vectors on the shared face $\mathcal{F}$, $\mathbf{n}^+$ and $\mathbf{n}^-$. We have the natural relation $\mathbf{n}^+ = -\mathbf{n}^-$.

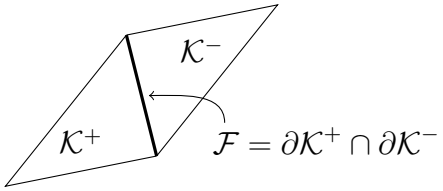

Figure 2: Two elements sharing a face

We denote the jump in a quantity q across the discontinuity at face $\mathcal{F}$ by $[\![q]\!]$ and the average value of the quantity at the discontinuity by $\{\!\{q\}\!\}$. We denote by q^+ the value of $q(\mathbf{x})$ at $\mathbf{x} \in \mathcal{F}$ when the value of q in element $\mathcal{K}^+$ is used, and similar for q^- when the value in $\mathcal{K}^-$ is used. On internal faces $\mathcal{F} = \partial\mathcal{K}^+ \cap \partial\mathcal{K}^-$ we define the jump and average

operators as

$$[\![q]\!] = a^+ - a^- \qquad \text{and} \qquad \{\!\{q\}\!\} = \frac{1}{2}(a^+ + a^-). \tag{7}$$

At the boundary of the computational domain $\mathbf{x} \in \partial\Omega$ we define

$$[\![q]\!] = q^+ \qquad \text{and} \qquad \{\!\{q\}\!\} = \frac{1}{2}q^+. \tag{8}$$

We will always denote the element that shares a face with the domain boundary as $\mathcal{K}^+$.

In the derivations of the weak form of the governing equations we will need to use the rules for integration by parts in multiple dimensions, repeated here for ease of reference:

$$\int_\Omega \mathbf{a} \cdot \nabla b \,\mathrm{d}\mathbf{x} = \int_{\partial\Omega} \mathbf{a} b \cdot \mathbf{n} \,\mathrm{d}s - \int_\Omega (\nabla \cdot \mathbf{a}) b \,\mathrm{d}\mathbf{x}, \tag{9}$$

$$\int_\Omega (\nabla \cdot \mathbf{a})\, b \,\mathrm{d}\mathbf{x} = \int_{\partial\Omega} \mathbf{a} b \cdot \mathbf{n} \,\mathrm{d}s - \int_\Omega \mathbf{a} \cdot \nabla b \,\mathrm{d}\mathbf{x}. \tag{10}$$

3.2 Weak form of the Navier-Stokes equations

The variational, or weak, form of the governing equations is created by the normal finite element methodology: we multiply by a test function $\mathbf{v}$ and integrate over the domain. Below we treat term by term in the equations.

The time derivative is converted to weak form in a straightforward manner when we use a second order backward difference formulation, BDF2, for the time discretisation. We get

$$\int_\Omega \frac{1}{2\Delta t} \left[3(\rho\mathbf{u})^{k+1} - 4(\rho\mathbf{u})^k + (\rho\mathbf{u})^{k-1}\right] \cdot \mathbf{v} \,\mathrm{d}x. \tag{11}$$

Other terms in the equations which does not include spatial derivatives of the unknown functions, such as the gravity term, are treated in the same way.

The convective term is in the continuous Galerkin method treated similarly to the time derivative and we hence get simply

$$\int_\Omega \nabla \cdot (\rho\mathbf{u} \otimes \mathbf{u}^C) \cdot \mathbf{v} \,\mathrm{d}x. \tag{12}$$

Here the $\otimes$ operator is used to denote the outer product of two vectors. This produces a tensor. Later we will use the inner product of a tensor and a vector which is denoted : and produces a scalar. To linearise the convective term we use $\mathbf{u}^C$ defined in equation (6).

If the computational mesh cannot adequately capture sharp gradients in the solution then stabilisation terms must be added in the continuous Galerkin method. The most well

known method is to add diffusion in the streamwise direction which can be accomplished by the streamline upwind Petrov-Galerkin method, SUPG [3, 13]. In this work we will ensure that the Peclet number giving the element wise balance of diffusion and convection is such that no stabilisation is needed.

To derive the discontinuous Galerkin method for the convective term we start with the need to establish a link between the values in neighbouring elements. We do this by splitting the integral over the domain into integrals over each element and performing integration by parts which gives us

$$\int_\Omega \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}^C) \cdot \mathbf{v} \, \mathrm{d}x = \sum_{\mathcal{K} \in \mathcal{T}} \int_{\mathcal{K}} \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}^C) \cdot \mathbf{v} \, \mathrm{d}x = \sum_{\mathcal{K} \in \mathcal{T}} \int_{\partial \mathcal{K}} \mathbf{n} \cdot (\rho \mathbf{u} \otimes \mathbf{u}^C) \cdot \mathbf{v} \, \mathrm{d}s - \sum_{\mathcal{K} \in \mathcal{T}} \int_{\mathcal{K}} (\rho \mathbf{u} \otimes \mathbf{u}^C) : \nabla \mathbf{v} \, \mathrm{d}x. \tag{13}$$

The sum of surface integrals in equation (13) will contain the integral over each internal face twice, once for each element that shares a face $\mathcal{F} = \partial \mathcal{K}^+ \cap \partial \mathcal{K}^-$. If we select the reference normal on the face to be $\mathbf{n}^+$ we can write the contribution from the two integrals as a jump since $\mathbf{n}^+ = -\mathbf{n}^-$ and get

$$\sum_{\mathcal{K} \in \mathcal{T}} \int_{\mathcal{F} \in \partial \mathcal{K}} \mathbf{n} \cdot (\rho \mathbf{u} \otimes \mathbf{u}^C) \cdot \mathbf{v} \, \mathrm{d}s = \sum_{\mathcal{F} \in \Gamma_{IND}} \int_{\mathcal{F}} \mathbf{n}^+ \cdot [\![(\rho \mathbf{u} \otimes \mathbf{u}^C) \cdot \mathbf{v}]\!] \, \mathrm{d}s. \tag{14}$$

We here denote by Γ_I the set of internal inter-element faces, by Γ_D the external faces with Dirichlet boundary conditions and by Γ_N the external faces with Neumann boundary conditions. On the boundary $\partial\Omega = \Gamma_D \cup \Gamma_N \equiv \Gamma_{DN}$ we always refer to the boundary cell as $\mathcal{K}^+$ which will, in addition to the definition in equation (8), lead to the rewrite in terms of the jump in equation (14) being valid also on the boundary faces.

We link the discontinuous values in the elements by requiring that the flux $\mathbf{F} = \rho \mathbf{u} \otimes \mathbf{u}$ is single valued at the shared face $\mathcal{F}$. This ensures local conservation [5]. We must chose a flux that is *consistent*, i.e. it should reduce to the continuous value if there is no jump at the interface.

We define a blended flux that uses a blending parameter β to blend between the upwind and downwind fluxes. To do this we define an upwind and a downwind convecting velocity in the direction of the normal $\mathbf{n}$ as

$$u_U = \frac{1}{2}(\mathbf{u}^C \cdot \mathbf{n} + |\mathbf{u}^C \cdot \mathbf{n}|) \tag{15}$$

$$u_D = \frac{1}{2}(\mathbf{u}^C \cdot \mathbf{n} - |\mathbf{u}^C \cdot \mathbf{n}|). \tag{16}$$

Note that either u_U^+ or u_D^+ will be zero and vice versa for the values in $\mathcal{K}^-$. Note also that if we define the normal direction to be $\mathbf{n}^+$ we can write $u_U^+ + u_U^- = [\![u_U]\!]$ where the

superscripts on u_U on the left hand side indicate only which element $\mathbf{u}^C$ is taken from and the normal is in both cases $\mathbf{n}^+$.

We will use a single valued flux $\mathrm{F}_{\mathbf{n}+}$ which we define in the direction of the normal $\mathbf{n}^+$. This means that we can take this flux out of the jump term in equation (14) and write

$$\sum_{\mathcal{F}\in\Gamma_{IND}} \int_{\mathcal{F}} \mathbf{n}^+ \cdot [\![(\rho\mathbf{u}\otimes\mathbf{u}^C)\cdot\mathbf{v}]\!]\,\mathrm{d}s = \sum_{\mathcal{F}\in\Gamma_{IND}} \int_{\mathcal{F}} \mathrm{F}_{\mathbf{n}+}\cdot[\![\mathbf{v}]\!]\,\mathrm{d}s. \tag{17}$$

We define the blended flux in terms of the blending factor β and the upwind and downwind convecting velocities. We take $\beta = 0$ to be a pure upwind flux and $\beta = 1$ to be a pure downwind flux and define

$$\mathrm{F}_{\mathbf{n}+} = (1-\beta)[\![\rho\mathbf{u}u_U]\!] + \beta[\![\rho\mathbf{u}u_D]\!]. \tag{18}$$

The diffusive term is treated by standard integration by parts in the continuous Galerkin method. This removes the need for the basis functions describing $\mathbf{u}$ to be two times differentiable. The weak form is then

$$\int_\Omega \nabla\cdot\mu\left[(\nabla\mathbf{u})+(\nabla\mathbf{u})^T\right]\cdot\mathbf{v}\,\mathrm{d}x = \int_{\partial\Omega}\mathbf{n}\cdot\mu\left[(\nabla\mathbf{u})+(\nabla\mathbf{u})^T\right]\cdot\mathbf{v}\,\mathrm{d}x - \int_\Omega \mu\left[(\nabla\mathbf{u})+(\nabla\mathbf{u})^T\right]:\nabla\mathbf{v}\,\mathrm{d}x. \tag{19}$$

In the discontinuous Galerkin method much work has gone into treating elliptic equations like the diffusive term in the Navier-Stokes equations. Elliptic fluxes have no inherent direction, so methods like up-winding are not applicable. The most commonly used method to establish a suitable weak form is the symmetric interior penalty Galerkin method, SIPG, by by Douglas Arnold [1]. We will not need to solve equations involving elliptic terms with the discontinuous Galerkin method in this presentation, so we will not discuss such methods further.

The pressure term can be written in two ways, either directly or after integration by parts. The latter is valid also if the pressure is approximated by piecewise constants, since we do not need to take the spatial derivative of the pressure function. The two forms are

$$\int_\Omega \nabla p\cdot\mathbf{v}\,\mathrm{d}x \qquad \text{and} \qquad \int_{\partial\Omega} p\mathbf{n}\cdot\mathbf{v}\,\mathrm{d}s - \int_\Omega p\nabla\cdot\mathbf{v}\,\mathrm{d}x. \tag{20}$$

The divergence free criterion is multiplied with a scalar test function q that belongs to the same function space as the pressure and we get

$$\int_\Omega \nabla\cdot\mathbf{u}\;q\,\mathrm{d}x. \tag{21}$$

This ensures that the divergence is zero in the space of the pressure function, but does not ensure local conservation in each element unless the pressure function space contains functions that are piecewise constant in each element.

3.3 Summary

To solve the coupled Navier-Stokes equations written on weak form for the continuous Galerkin method we must find the unknown functions $\mathbf{u}$ and p such that for all test functions $\mathbf{v}$ and q we have

$$\int_\Omega \frac{1}{2\Delta t}\left[3(\rho\mathbf{u})^{k+1} - 4(\rho\mathbf{u})^k + (\rho\mathbf{u})^{k-1}\right]\cdot\mathbf{v}\,\mathrm{d}x + \int_\Omega \nabla\cdot(\rho\mathbf{u}\otimes\mathbf{u}^C)\cdot\mathbf{v}\,\mathrm{d}x + \int_\Omega \mu\left[(\nabla\mathbf{u}) + (\nabla\mathbf{u})^T\right] : \nabla\mathbf{v}\,\mathrm{d}x - \int_{\partial\Omega}\mathbf{n}\cdot\mu\left[(\nabla\mathbf{u}) + (\nabla\mathbf{u})^T\right]\cdot\mathbf{v}\,\mathrm{d}x + \int_\Omega p\nabla\cdot\mathbf{v}\,\mathrm{d}x - \int_{\partial\Omega} p\mathbf{n}\cdot\mathbf{v}\,\mathrm{d}s - \int_\Omega \rho\mathbf{g}\cdot\mathbf{v}\,\mathrm{d}x + \int_\Omega \nabla\cdot\mathbf{u}\; q\,\mathrm{d}x = 0 \tag{22}$$

To solve the advection equation for the colour function using the discontinuous Galerkin method we must find the unknown function α such that for all test functions w we have

$$\sum_{\mathcal{K}\in\mathcal{T}}\int_{\mathcal{K}}\frac{1}{2\Delta t}\left[3\alpha^{k+1} - 4\alpha^k + \alpha^{k-1}\right]w\,\mathrm{d}x + \sum_{\mathcal{F}\in\Gamma_{IND}}\int_{\mathcal{F}} F_\alpha[\![w]\!]\,\mathrm{d}s - \sum_{\mathcal{K}\in\mathcal{T}}\int_{\mathcal{K}}(\alpha\mathbf{u}^C)\cdot\nabla w\,\mathrm{d}x = 0, \tag{23}$$

where the flux F_α is defined as $F_\alpha = (1-\beta)[\![\alpha u_U]\!] + \beta[\![\alpha u_D]\!]$ and u_U and u_D are the upstream and downstream velocities in the normal direction as defined in equations (15) and (16).

4 FREE SURFACE EVOLUTION

Several popular methods for treating flows of two immiscible fluids with a free surface exist. Very high accuracy can be obtained by deforming the mesh to follow the interface with an arbitrary Lagrangian-Eulerian method [11], but this cannot handle complex interface topology changes such as merging or splitting of the interface without remeshing. Such large deformations will occur e.g. with an overturning surface wave. The most commonly used methods for handling general interfaces are the level set method [20] and the volume of fluid, VOF, method [12]. In the VOF method the interface is tracked by a scalar that is 1.0 in one fluid phase and 0.0 in the other. The behaviour of the scalar function, know as the colour function is governed by a transport equation,

$$\frac{\partial\alpha}{\partial t} + \mathbf{u}\cdot\nabla\alpha = 0. \tag{24}$$

The fluid properties such as the kinematic viscosity ν and the density ρ are calculated from the colour function field α with a simple weighting formula. E.g. the density is calculated from the density in fluid 1, ρ_1 and the density in fluid 2, ρ_2 as

$$\rho = \rho_1\alpha + \rho_2(1-\alpha). \tag{25}$$

We will apply an algebraic volume of fluid method, the high resolution interface capturing scheme, HRIC [19]. Algebraic volume of fluid methods work by adjusting the parameter β in equation (18) to keep the interface sharp. Such methods are discussed next, first in general and then the specifics of the HRIC method are shown.

4.1 Preliminaries

To study algebraic VOF methods we look at the convection of a scalar function α from a central control volume $\mathcal{K}_\mathrm{C}$ to a downstream control volume $\mathcal{K}_\mathrm{D}$. The control volumes are separated by the face $\mathcal{F}$. Upstream of the central control volume there is an upstream control volume $\mathcal{K}_\mathrm{U}$. We stay within the framework of a zeroth order approximation of the scalar function, either by the finite volume method or a zeroth order discontinuous Galerkin finite element method. The situation is sketched in figure 3.

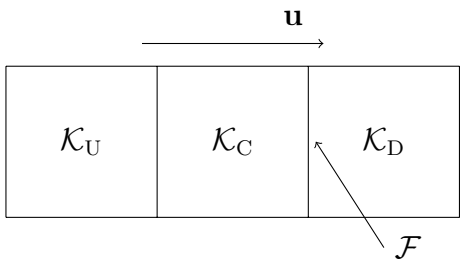

Figure 3: Three neighbouring control volumes, upwind, central and downwind

We look at the situation where the scalar function α is increasing from C to D, see figure 4. To satisfy the convective boundedness criterion, CBC, we must require that $\alpha_\mathrm{C} \leq \alpha_\mathcal{F} \leq \alpha_\mathrm{D}$, see [7]. We introduce the normalised quantities

$$\tilde{\alpha_\mathrm{C}} = \frac{\alpha_\mathrm{C} - \alpha_\mathrm{U}}{\alpha_\mathrm{D} - \alpha_\mathrm{U}} \qquad \text{and} \qquad \tilde{\alpha_\mathcal{F}} = \frac{\alpha_\mathcal{F} - \alpha_\mathrm{U}}{\alpha_\mathrm{D} - \alpha_\mathrm{U}}, \tag{26}$$

and by eliminating the upwind value α_U we get

$$\alpha_\mathcal{F} = (1-\beta)\alpha_\mathrm{C} + \beta\alpha_\mathrm{D} \qquad \text{and} \qquad \beta = \frac{\tilde{\alpha_\mathcal{F}} - \tilde{\alpha_\mathrm{C}}}{1 - \tilde{\alpha_\mathrm{C}}}. \tag{27}$$

We see from equation (27) that β is a downwinding factor. For $\tilde{\alpha_\mathcal{F}} = 1$ we get $\beta = 1$ and we have a pure downwind flux. If we take $\tilde{\alpha_\mathcal{F}} = \tilde{\alpha_\mathrm{C}}$ then we get $\beta = 0$ and we have a pure upwind flux.

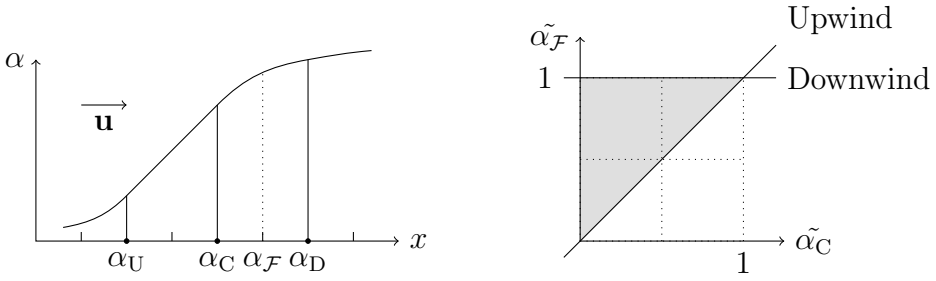

Figure 4: The 1D scalar function $\alpha_{\mathcal{F}}$ (left) the Normalised Variable Diagram (right)

The normalized variable diagram by Leonard [16] in figure 4 shows the region where face value $\alpha_{\mathcal{F}}$ satisfies the convection boundedness criterion, CBC, in the $\tilde{\alpha_{\mathrm{C}}}$-$\tilde{\alpha_{\mathcal{F}}}$ plane. For $\tilde{\alpha_{\mathrm{C}}} \notin [0,1]$ only the upwind scheme is stable, while for $\tilde{\alpha_{\mathrm{C}}} \in [1,0]$ we can select to blend the upwind scheme and the downwind scheme.

4.2 High Resolution Interface Capturing

The HRIC scheme [19] improves on the diffusiveness of the unconditionally stable upwind scheme by calculating the flux $\alpha_{\mathcal{F}}$ based on a CBC-compliant scheme that maximizes the downwind contribution which contributes to sharpening the interface. As can be seen in figure 5 the HRIC scheme follows the upper boundary of the TVD region in the normalised variable diagram.

$$\tilde{\alpha}_{\mathcal{F}} = \begin{cases} \tilde{\alpha}_{\mathrm{C}} & \text{if} \quad \tilde{\alpha}_{\mathrm{C}} \notin [0,1] \\ 2\tilde{\alpha}_{\mathrm{C}} & \text{if} \quad \tilde{\alpha}_{\mathrm{C}} \in [0, \frac{1}{2}] \\ 1 & \text{if} \quad \tilde{\alpha}_{\mathrm{C}} \in [\frac{1}{2}, 1] \end{cases} \tag{28}$$

Equation (28) is the same as the HYPER-C scheme by Leonard [17] with Courant number equal to 0.5. The C in HYPER-C is for compressive and Leonard shows that the heavily downwind biased HYPER-C scheme will reduce all gradients to step functions. The VOF interface we seek is indeed a step function, but the problem with heavy sharpening is that it will align the free surface with the mesh and the true free surface is not in general aligned with the mesh. Too heavy sharpening causes stair-casing and unphysical wiggles of the interface. [22]

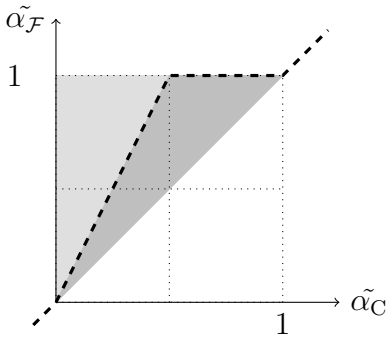

Figure 5: The HRIC scheme shown in the normalised variable diagram

In the HRIC scheme the problem with downwind sharpening causing alignment of the interface to the mesh is mitigated by introducing more upwinding when the angle θ between the face normal $\mathbf{n}_{\mathcal{F}}$ and the interface normal $\mathbf{n}_\alpha$ is large by blending with the upwind value

$$\tilde{\alpha}^*_{\mathcal{F}} = \gamma_{\mathbf{n}}\tilde{\alpha}_{\mathcal{F}} + (1-\gamma_{\mathbf{n}})\tilde{\alpha}_{\mathrm{C}} \qquad \gamma_{\mathbf{n}} = \sqrt{\cos(\theta)}. \tag{29}$$

Equations (28) and (29) may cause convergence problems if the Courant number Co is too large. Even the original HYPER-C scheme, which is designed to be as close to the downwind scheme as possible while still being bounded, is less compressive than equation (28) for Courant numbers higher than 0.5. In the HRIC scheme the Courant number dependency is as follows to ensure stability:

$$\tilde{\alpha}^{**}_{\mathcal{F}} = \begin{cases} \tilde{\alpha}^*_{\mathcal{F}} & \text{if} \quad \mathrm{Co} < 0.3 \\ \tilde{\alpha}_{\mathrm{C}} + (\tilde{\alpha}^*_{\mathcal{F}} - \tilde{\alpha}_{\mathrm{C}})\frac{0.7-\mathrm{Co}}{0.7-0.3} & \text{if} \quad 0.3 < \mathrm{Co} < 0.7 \\ \tilde{\alpha}_{\mathrm{C}} & \text{if} \quad \mathrm{Co} > 0.7 \end{cases} \tag{30}$$

Having calculated $\tilde{\alpha}^{**}_{\mathcal{F}}$ we can use equation (27) to calculate β and this can be used to formulate the DG-FEM convection term in the transport equation for α.

One issue with using methods based on Leonard's normalised variable diagram to calculate β is the dependency of $\tilde{\alpha_{\mathrm{C}}}$ on α_{U}. On a general unstructured mesh the upstream cell value is not necessarily available. Having tested several methods for establishing an approximate upstream value the method by Ubbink [22] was found to give good results. In our HRIC implementation the upstream value is calculated from:

$$\alpha_{\mathrm{U}} = \alpha_{\mathrm{D}} - 2(\nabla\alpha)_{\mathrm{C}} \cdot \mathbf{d} \tag{31}$$

Here $\mathbf{d}$ is the vector from the cell centre of $\mathcal{K}_{\mathrm{C}}$ to the cell centre of $\mathcal{K}_{\mathrm{D}}$. We calculate $(\nabla\alpha)_{\mathrm{C}}$ which is the gradient of the colour function in the centre cell by a least squares gradient reconstruction procedure [24] taking into account all cells that share one or more vertices with the centre cell.

5 RESULTS

5.1 Diagonal advection of a square

Transport of a 2D 0.5×0.5 square in a velocity field $\mathbf{u} = [1, 1]$ for time $t \in [0, 0.5]$ and $\mathbf{u} = [-1, -1]$ for $t \in [0.5, 1.0]$ is used as a test case to test the implemented VOF scheme. At the end of the simulation the square should end up in the same location as it started and the L_2 error between the initial and final α fields is used as a benchmark to test the method.

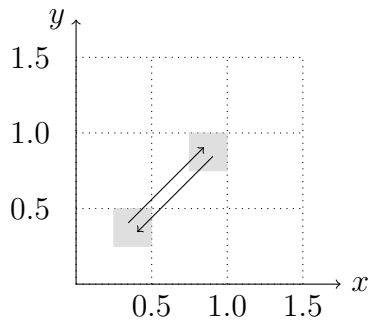

Figure 6: Initial location (bottom left) and path travelled by the square

Both the HRIC convection scheme and a pure upwind scheme are tested. The HRIC scheme manages to keep the interface sharper from a pure visual inspection point of view and also has lower L_2 errors at the final time step both in terms of absolute errors and convergence rate. The results are presented in table 1. The mesh employed is a regular mesh of $N \times N$ squares covering $x, y \in [0, 1.5]$ where each square is made into two triangles with alternating left and right pointing diagonals. The convergence rate k is defined as the $\log_2$ of the error at the courser mesh divided by the the error at the mesh with one refinement level more. Discontinuous Galerkin elements of order 0 are used to approximate the colour function α, i.e piecewise constants.

The HRIC scheme conserves the mass in terms of the integrated value of the colour function, while the upwind scheme looses some mass due to transport out of the domain in the upper right corner.

5.2 Taylor-Green vortex

The Taylor-Green vortex [9] is a much used benchmark for single fluid Navier-Stokes solution methods as it gives an analytical solution in both space and time. The solution is a doubly periodic vortex that decays with time. In 2D the solution on a periodic square

Table 1: VOF error norm and convergence rate k for a set of refined meshes with $2N^2$ elements.

Method	N	L_2 error	k
Upwind	32	1.000	-
Upwind	64	0.712	0.49
Upwind	128	0.505	0.50
HRIC	32	0.370	-
HRIC	64	0.197	0.91
HRIC	128	0.107	0.89

domain $x, y \in [0, 2]$ reads

$$u_x = -\sin(\pi y)\cos(\pi x)\exp(-2\pi^2\nu t), \tag{32}$$
$$u_y = \sin(\pi x)\cos(\pi y)\exp(-2\pi^2\nu t), \tag{33}$$
$$p = \frac{1}{4}(\cos(2\pi x) + \cos(2\pi y))\exp(-4\pi^2\nu t). \tag{34}$$

In a periodic domain there are no Dirichlet boundary conditions for the pressure field, which is only determined up to an arbitrary constant. We set the spatial averaged pressure to be zero as an additional constraint. This null space of the pressure is removed by a method implemented in the linear algebra library PETSc [2] for the segregated solver and with a Lagrange multiplier for the coupled solver.

Results are calculated both with a coupled and an IPCS segregated continuous Galerkin FEM solver. The standard P2P1 Taylor-Hood elements are used which are continuous Galerkin elements with second order polynomials for the velocities and first order polynomials for the pressure. The viscosity is $\nu = 0.01$ and the density is $\rho = 1.0$.

The error after $t = 1.0$ are reported for a variety of mesh sizes. A fine enough time step of 0.01 s is used along with 10 inner iterations in the IPCS solver to avoid time integration errors influencing the spatial convergence. A set of regular meshes with alternating left and right diagonals are used. The number of elements along each side is 8, 16, 24, 32 and 40. The results can be seen in tables 2 and 3. Except for the time needed to solve the problem the tables are virtually identical. The expected convergence rate of $N + 1$ is observed, where N is the polynomial degree of the approximating functions.

The result tables also show the time needed to solve the problem on one CPU on a desktop machine. It should be noted that the implemented solver is not optimised in any way except for choosing appropriate linear solvers and preconditioners. Examples of much faster Navier-Stokes solvers implemented in FEniCS exist [18]. The relative difference between the methods should still be somewhat representative.

In the segregated IPCS Navier-Stokes solver the momentum prediction equation and the velocity correction equations are solved with the PETSc generalised minimal residual solver [2] with the additive Schwarz preconditioner. The pressure Poisson equation is

Table 2: Spatial convergence with the IPCS solver

Discr.	Errors in L_2 norm			$\mathcal{O}\left(h^k\right)$			Duration
h	u_x	u_y	p	u_x	u_y	p	wallclock
0.35355	8.70×10^{-2}	8.70×10^{-2}	2.25×10^{-1}				6.1s
0.17678	4.44×10^{-3}	4.44×10^{-3}	5.10×10^{-2}	4.29	4.29	2.14	11.1s
0.11785	1.07×10^{-3}	1.07×10^{-3}	2.30×10^{-2}	3.51	3.51	1.97	20.0s
0.08839	4.35×10^{-4}	4.35×10^{-4}	1.30×10^{-2}	3.14	3.14	1.99	32.6s
0.07071	2.20×10^{-4}	2.20×10^{-4}	8.30×10^{-3}	3.04	3.04	2.00	48.8s
Discr.	Errors in H^1 norm			$\mathcal{O}\left(h^k\right)$			Duration
h	u_x	u_y	p	u_x	u_y	p	wallclock
0.35355	1.24×10^{0}	1.24×10^{0}	6.10×10^{-1}				6.1s
0.17678	1.65×10^{-1}	1.65×10^{-1}	1.60×10^{-1}	2.92	2.92	1.93	11.1s
0.11785	6.14×10^{-2}	6.14×10^{-2}	9.11×10^{-2}	2.43	2.43	1.39	20.0s
0.08839	3.26×10^{-2}	3.26×10^{-2}	6.40×10^{-2}	2.20	2.20	1.23	32.6s
0.07071	2.04×10^{-2}	2.04×10^{-2}	4.97×10^{-2}	2.10	2.10	1.14	48.8s

Table 3: Spatial convergence with the coupled solver

Discr.	Errors in L_2 norm			$\mathcal{O}\left(h^k\right)$			Duration
h	u_x	u_y	p	u_x	u_y	p	wallclock
0.35355	8.70×10^{-2}	8.70×10^{-2}	2.25×10^{-1}				3.1s
0.17678	4.44×10^{-3}	4.44×10^{-3}	5.10×10^{-2}	4.29	4.29	2.14	10.4s
0.11785	1.07×10^{-3}	1.07×10^{-3}	2.30×10^{-2}	3.51	3.51	1.97	25.5s
0.08839	4.35×10^{-4}	4.35×10^{-4}	1.30×10^{-2}	3.14	3.14	1.99	52.7s
0.07071	2.20×10^{-4}	2.20×10^{-4}	8.30×10^{-3}	3.04	3.04	2.00	1m 53.1s
Discr.	Errors in H^1 norm			$\mathcal{O}\left(h^k\right)$			Duration
h	u_x	u_y	p	u_x	u_y	p	wallclock
0.35355	1.24×10^{0}	1.24×10^{0}	6.11×10^{-1}				3.1s
0.17678	1.65×10^{-1}	1.65×10^{-1}	1.60×10^{-1}	2.92	2.92	1.93	10.4s
0.11785	6.14×10^{-2}	6.14×10^{-2}	9.11×10^{-2}	2.43	2.43	1.39	25.5s
0.08839	3.26×10^{-2}	3.26×10^{-2}	6.40×10^{-2}	2.20	2.20	1.23	52.7s
0.07071	2.04×10^{-2}	2.04×10^{-2}	4.97×10^{-2}	2.10	2.10	1.14	1m 53.1s

solved with the PETSc minimal residual solver preconditioned with the hypre Boomer-AMG algebraic multigrid preconditioner [6]. The coupled Navier-Stokes solver uses the PETSc built in LU-solver.

For showing convergence in time a fine mesh of 200×200 squares divided by alternating left and right diagonals is employed. The number of IPCS inner iterations employed is 20 and the simulation is run until $t = 6.0$. The results can be seen in tables 4 and 5. Both implemented solvers can be seen to be second order in time in the L_2 norm for velocities. The pressure convergence is not as uniform as the velocities, but it is also approximately second order.

The divergence free properties of the calculated velocity fields varies significantly between the two implemented Navier-Stokes solvers. As can be seen from the result tables the H^1 errors are of comparable magnitude, but if we study the maximum divergence calculated in the function space of the pressure, where it should be smallest, the error $||(\nabla \cdot u)_{V_p}||_{L_\infty}$, is still as much as ten orders of magnitude larger in the IPCS formulation compared to the coupled formulation. This difference is reduced significantly as the time step is reduced. The balance between time step and divergence will favour the IPCS solver when the problem size grows as the coupled solver is significantly less efficient than the IPCS solver in terms of CPU time needed to solve the same size problems.

To be able to apply the presented DG-FEM scheme for advection of a VOF colour function the divergence must be low in the function space of the advected colour function. Both implemented solvers have significantly higher divergence as seen in DG0, the DG-FEM space of piecewise constants, compared to the P1 function space of the pressure, continuous piecewise linear functions. On the Taylor-Green test case the maximum divergence is for one example 10^{-13} in P1 and in DG0 in the same example it is 10^{-5}. Ideally the divergence free properties should be such that the divergence of the velocity is close to machine epsilon in both the function space of the pressure and in that of the colour function.

Table 4: Temporal convergence with the IPCS solver

Timestep	Errors in L_2 norm			$\mathcal{O}\left(\Delta t^k\right)$			Duration
Δt	u_x	u_y	p	u_x	u_y	p	wallclock
2.000	7.78×10^{-2}	7.78×10^{-2}	3.81×10^{-1}				9m 35.0s
1.000	1.85×10^{-2}	1.85×10^{-2}	8.39×10^{-2}	2.07	2.07	2.18	13m 24.4s
0.500	3.88×10^{-3}	3.88×10^{-3}	1.82×10^{-2}	2.26	2.26	2.21	15m 11.2s
0.250	9.77×10^{-4}	9.77×10^{-4}	4.21×10^{-3}	1.99	1.99	2.11	16m 19.5s
0.125	2.42×10^{-4}	2.42×10^{-4}	7.89×10^{-4}	2.01	2.01	2.42	18m 7.2s
Timestep	Errors in H^1 norm			$\mathcal{O}\left(\Delta t^k\right)$			Duration
Δt	u_x	u_y	p	u_x	u_y	p	wallclock
2.000	1.45×10^{-1}	1.45×10^{-1}	1.15×10^{-1}				9m 35.0s
1.000	2.90×10^{-2}	2.90×10^{-2}	3.23×10^{-2}	2.32	2.32	1.84	13m 24.4s
0.500	5.57×10^{-3}	5.57×10^{-3}	1.96×10^{-2}	2.38	2.38	0.72	15m 11.2s
0.250	1.44×10^{-3}	1.44×10^{-3}	1.36×10^{-2}	1.95	1.95	0.53	16m 19.5s
0.125	4.03×10^{-4}	4.09×10^{-4}	3.46×10^{-3}	1.84	1.82	1.97	18m 7.2s

Table 5: Temporal convergence with the coupled solver

Timestep	Errors in L_2 norm			$\mathcal{O}\left(\Delta t^k\right)$			Duration
Δt	u_x	u_y	p	u_x	u_y	p	wallclock
2.000	6.53×10^{-2}	6.53×10^{-2}	3.59×10^{-1}				10m 6.5s
1.000	1.62×10^{-2}	1.62×10^{-2}	7.96×10^{-2}	2.01	2.01	2.17	19m 57.6s
0.500	3.97×10^{-3}	3.97×10^{-3}	1.83×10^{-2}	2.03	2.03	2.12	38m 7.8s
0.250	9.77×10^{-4}	9.77×10^{-4}	4.18×10^{-3}	2.02	2.02	2.13	64m 9.2s
0.125	2.42×10^{-4}	2.42×10^{-4}	7.81×10^{-4}	2.01	2.01	2.42	127m 8.1s
Timestep	Errors in H^1 norm			$\mathcal{O}\left(\Delta t^k\right)$			Duration
Δt	u_x	u_y	p	u_x	u_y	p	wallclock
2.000	9.10×10^{-2}	9.10×10^{-2}	1.07×10^{-1}				10m 6.5s
1.000	2.26×10^{-2}	2.26×10^{-2}	2.37×10^{-2}	2.01	2.01	2.17	19m 57.6s
0.500	5.53×10^{-3}	5.53×10^{-3}	5.62×10^{-3}	2.03	2.03	2.08	38m 7.8s
0.250	1.37×10^{-3}	1.37×10^{-3}	1.81×10^{-3}	2.01	2.01	1.64	64m 9.2s
0.125	3.67×10^{-4}	3.68×10^{-4}	1.33×10^{-3}	1.90	1.89	0.45	127m 8.1s

6 CONCLUSIONS AND FURTHER WORK

We have described a work in progress to create a finite element multi-phase flow solver. The current state of the solution method is presented along with some preliminary results. The main focus of the work has been to learn about the challenges of simulating realistic non-linear two-phase flows with finite element methods. We have shown that we can produce satisfactory results for advection of the VOF colour function and order optimal solutions of the Navier-Stokes equations for single phase flow.

We have identified shortcomings that prevent us from providing trustworthy results for two-phase flows. Foremost among these is that we must improve mass conservation properties by reducing the divergence of the calculated velocity field in the function space of the colour function. A further possible refinement will be to improve handling of the integral equations near the free surface so that we can treat the free surface discontinuity explicitly and hence avoid non-physical momentum exchange between the two phases. Such non-physical momentum exchange will happen when we compute integrals across the discontinuity in the free surface zone by using element average values for the density and viscosity. We will also investigate methods to improve the free surface capturing method and evaluate more advanced volume of fluid or level set methods to obtain a sharp interface and at the same time ensure that the mass of each phase is conserved.

ACKNOWLEDGEMENTS

Mikael Mortensen is supported through a center of excellence grant from the research council of Norway to the Center for Biomedical Computing at Simula Research Laboratory.

REFERENCES

[1] D. N. Arnold. An interior penalty finite element method with discontinuous elements. *SIAM journal on numerical analysis*, 19(4):742–760, 1982.

[2] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, and H. Zhang. PETSc users manual. Technical Report ANL-95/11 - Revision 3.5, Argonne National Laboratory, 2014.

[3] A. N. Brooks and T. J. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Computer methods in applied mechanics and engineering*, 32(1):199–259, 1982.

[4] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Mathematics of computation*, 22(104):745–762, 1968.

[5] B. Cockburn. Discontinuous Galerkin methods. *ZAMM*, 83(11):731–754, 2003.

[6] R. D. Falgout and U. M. Yang. hypre: A library of high performance preconditioners. In *Computational science - ICCS 2002*, pages 632–641. Springer, 2002.

[7] P. H. Gaskell and A. K. C. Lau. Curvature-compensated convective transport: SMART, a new boundedness- preserving transport algorithm. *International journal for numerical methods in fluids*, 8(6):617–641, 1988.

[8] K. Goda. A multistep technique with implicit difference schemes for calculating two- or three-dimensional cavity flows. *Journal of computational physics*, 30(1):76–95, 1979.

[9] A. Green and G. Taylor. Mechanism of the production of small eddies from larger ones. In *Proceedings of the royal society of London. Series A, mathematical and physical sciences*, volume 158, pages 499–521, 1937.

[10] J. S. Hesthaven and T. Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*, volume 54. Springer, 2007.

[11] C. Hirt, A. A. Amsden, and J. Cook. An arbitrary Lagrangian-Eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14(3):227–253, 1974.

[12] C. W. Hirt and B. D. Nichols. Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of computational physics*, 39(1):201–225, 1981.

[13] C. Johnson and U. Nävert. An analysis of some finite element methods for advection-diffusion problems. *North-Holland mathematics studies*, 47:99–116, 1981.

[14] O. A. Ladyzhenskaya. *The mathematical theory of viscous incompressible flow*, volume 76 of *Mathematics and its applications*. Gordon and Breach, 1969. Translated to English by Richard A. Silverman.

[15] T. Landet. The Ocellaris finite element solver for free surface flows, 2015. `http://trlandet.bitbucket.org/ocellaris/`.

[16] B. P. Leonard. Simple high-accuracy resolution program for convective modelling of discontinuities. *International journal for numerical methods in fluids*, 8(10):1291–1318, 1988.

[17] B. P. Leonard. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer methods in applied mechanics and engineering*, 88(1):17–74, 1991.

[18] M. Mortensen and K. Valen-Sendstad. Oasis: A high-level/high-performance open source Navier-Stokes solver. *Computer physics communications*, 188:177–188, 2015.

[19] S. Muzaferija, M. Peric, P. Sames, and T. Schellin. A two-fluid Navier-Stokes solver to simulate water entry. In *Proceedings from the 22nd symposium on naval hydrodynamics*, pages 277–289, Washington, DC, 1998.

[20] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[21] R. Temam. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires (II). *Archive for rational mechanics and analysis*, 33(5):377–385, 1969.

[22] O. Ubbink. *Numerical prediction of two fluid systems with sharp interfaces*. PhD thesis, Imperial College, University of London, 1997.

[23] J. van Kan. A second-order accurate pressure-correction scheme for viscous incompressible flow. *SIAM journal on scientific and statistical computing*, 7(3):870–891, 1986.

[24] H. K. Versteeg and W. Malalasekera. *An introduction to computational fluid dynamics: the finite volume method*. Prentice Hall, 2 edition, 2007.